

特開昭63-178666 (公)

「1」にする。

このように、受信した圧縮コードに基づいて圧縮コードのまま長さ（ランレングス）をチェックして正しいラインか否かを判定し、正しいラインであれば受信した圧縮コードをメモリに留保することにより、圧縮コードを一旦伸長してランレングスをチェックして正しいラインか否かを判定する場合に比べて判定処理を高速で行なうことができるようになる。

このようにしてFAX面情報復元エリアに留保された受信印刷指示が与えられることによつて、留保されたMHコードをデコードしてプリンタ・ライン・ビット・プレーンPLBP（これについては後述する）を作成し、このプリンタ・ライン・ビット・プレーンPLBPのデータをプリンタに出力する。

そこで、まずFAX面情報復元エリアに留保された圧縮コードをバイトあるいはワード単位で取出してデコードし、プリンタの1ライン分のイメージデータを作成する処理について説明する。

プリンタ以外のプリンタを使用することも考慮して、フアクシミリ装置の類に合わせて、1つのエリアが1728×24（5148バイト×3）である（このエリアを「プレーン」と称する）3個のプレーンからなるプリンタ・ライン・ビット・プレーンPLBPを2個使用する。なお、プリンタ・ライン・ビット・プレーンPLBPを2個持つのはデコード処理とプリンタ出力処理とを並行動作できるようにするためである。

この場合、実際のプリンタ・ライン・ビット・プレーンPLBPのためのメモリ空間は例えば第14図に示すようなメモリ空間としている。これは、送信モードにおけるビット・マップ展開で説明したように、プリンタへのイメージデータの転送フォーマットとフアクシミリ装置との間の転送フォーマットが異なるので、送信時とは逆にフアクシミリ装置から受信した1ドットラインのデータをプリンタに転送する際のフォーマット合わせで格納するためである。

そこで、デコード及びプリンタ・ライン・ビ

このイメージデータの作成処理では、圧縮コードをデコードして1行（1ライン）分のイメージデータを作成して、メモリ（このメモリを上述した「プリンタ・ライン・ビット・プレーンPLBP」と称する）に格納する。

この場合、この実施例におけるプリンタは1行（1ライン）が24×1440ドット構成であるのに対して、フアクシミリ装置においては1行の長さが1728ドットであるため、第15図に示すようにプリンタ・ライン・ビット・プレーンPLBPよりも装置にデコード処理で得られるビット・プレーンの方が両端が各々144ドット分大きくなる。したがって、この144ドット分はデコードしたデータをそのままプリンタ・ライン・ビット・プレーンPLBPに展開して印字したときには欠けることになるが、この点については具体的説明は省略するがビット変換処理を行なうことによつて大幅な欠落を回避できる。

そして、ここでは、プリンタ・ライン・ビット・プレーンPLBPのエリアとしてはこの実施例の

ト・プレーンPLBPの作成処理について第15図を参照して説明する。

この作成処理においては、受信装置処理部から其情報を得た後、プリンタ・ライン・ビット・プレーンカウンタPLNCNTを「0」にリセットし、ドットライン数（0〜24-1）を示すためのライン・ホリゾンタル・ポインタLINHPを「0」にリセットする。

その後、更に1行（24ドットライン）が全て白（0）であることを示す、つまり当該ラインには黒（1）が存在することを示すためのプリンタ・ライン・ビット・プレーン・ブラック・イグジストフラグPLNBKを「0」（すべて白の状態）にセットし、プリンタ・ライン・ビット・プレーンPLBPをクリアする。

そして、1ドットラインのビット位置を示すためのライン・ビット・ポインタLINBPを「0」にリセットした後、白（ホワイト）圧縮コードをデコードするホワイトデコード処理に移行する。

このホワイトデコード処理でEOL及びエラーEOLのいずれもを検出しなければ、ライン・ビット・ポインタLINBTPに圧縮コードのランレングス(RL)分だけ加算し、EOLをカウントするEOLカウンタを「0」にリセットした後、黒(ブラック)圧縮コードをデコードするブラックデコード処理に移行する。

このブラックデコード処理でEOL及びエラーEOLのいずれもを検出しなければ、当該ラインには黒があるのでプリンタ・ライン・ビット・プレーン・ブラック・イグジットフラグPLNBKを「1」にセットする。

その後、ライン・ビット・ポインタLINBTPが「1728」以上か否かを判別し、ライン・ビット・ポインタLINBTPが「1728」未満であれば、所定の位置に「1」(黒)を配置するビットセット処理をした後、ライン・ビット・ポインタLINBTPに圧縮コードのランレングス(RL)分だけ加算して、ランレングスから -1 ($RL-1$)してその値が「0」か否かを判別

し、 $(RL-1) \neq 0$ でなければ再度ライン・ビット・ポインタLINBTPが「1728」以上か否かの判別処理に戻ってビットセットを繰返す。

そして、ライン・ビット・ポインタLINBTPが「1728」以上になったとき又は $(RL-1) = 0$ になったときには、1ドットラインの最終ビットに達しているのものでそのままホワイトデコード処理に戻る。

このように、ここでは圧縮コードをデコードしてランレングスを求め、白MHコードについては単にカウンタを進め、黒MHコードについてのみそのランレングスで示されるビット分だけプリンタ・ライン・ビット・プレーンPLBPの所定のビットに「1」(黒)を配置する処理を行なう。それによつて、白についてはビット配置を行なわなくて済み、前述した送信処理の場合と同様に処理速度の高速化を図ることができる。

そして、ホワイトデコード処理又はブラックデコード処理においてEOLを検出したときには、EOLカウンタをインクリメント(+1)した後、

またホワイトデコード処理又はブラックデコード処理においてエラーEOLを検出したときにはそのまま、EOLカウンタのカウント値が「2」すなわちRTC(頁の終り)か否かを判別する。

このとき、EOLカウンタのカウント値が「2」でなければ、つまり頁の終りでなければ、印字ピッチが3.85ライン/mmが7.7ライン/mmか(3.85/7.7)かを判別して、印字ピッチが3.85ライン/mmのときにはライン・ホリゾンタル・ポインタLINHZPを「+2」し、印字ピッチが7.7ライン/mmのときにはライン・ホリゾンタル・ポインタLINHZPを「+1」する。なお、3.85ライン/mmのときにライン・ホリゾンタル・ポインタLINHZPを「+2」するのは後述するようにビットセット処理で7.7ライン/mmに合わせるために2ドットライン分同じビットセットを行なうためである。

そして、ライン・ホリゾンタル・ポインタLINHZPが「24」か否かつまり24ドットライン(1行)分のデコードが終了したか否かを判別

して、24ドットライン分のデコードが終了していなければ再度ライン・ビット・ポインタLINBTPを「0」にリセットしてデコードを繰返し、24ドットライン分のデコードが終了していれば、プリンタ・ライン・ビット・プレーン・ブラック・イグジット・フラグPLNBKが「1」か否かを判別する。

このとき、プリンタ・ライン・ビット・プレーン・ブラック・イグジット・フラグPLNBKが「1」であれば、プリンタ・ライン・ビット・プレーンPLBPに展開したイメージデータをプリンタに出力し、「1」なければつまり「0」であれば、その行はすべて白であつて単にラインフールド(改行)を行なえば足りるので、プリンタヘッドコントロールコマンド「LF」をプリンタに出力する。

その後、プリンタ・ライン・ビット・プレーン・カウンタPLNCNTをインクリメント(+1)して、プリンタ・ライン・ビット・プレーン・カウンタPLNCNTのカウント値が最大値(MA

ス) になったか否か、すなわち1頁の最大印字行数になったか否かを判別する。

このとき、プリンタ・ライン・ビット・プレーン・カウンタPLCNTのカウンタ値が最大値でなく1頁の最大行数になっていなければ、当該用紙に未定印字できるので、プリンタ・ライン・ビット・プレーン・ブラック・イグジット・フラグPLNBLEKが「1」か否かを判別する。

そして、プリンタ・ライン・ビット・プレーン・ブラック・イグジット・フラグPLNBLEKが「1」のときにはプリンタ・ライン・ビット・プレーンPLBPに「1」が配置されているので、ライン・水平・ポイントLINHPを「0」にリセットし、更にプリンタ・ライン・ビット・プレーン・ブラック・イグジット・フラグPLNBLEKを「0」にリセットしてプリンタ・ライン・ビット・プレーンPLBPをクリアする処理に戻る。

また、プリンタ・ライン・ビット・プレーン・ブラック・イグジット・フラグPLNBLEKが

「1」でなければ、プリンタ・ライン・ビット・プレーンPLBPに「1」が配置されていないので、そのままライン・ビット・ポイントLINHPを「0」にリセットする処理に戻る。

このようにデコードした行についてプリンタ・ライン・ビット・プレーンPLBPに「1」を配置したか否かを保持しておくことによつて、印字動作の高速化を図れると共に、ビット展開の処理の高速化を図ることができる。

なお、EOLカウンタEOLCNTのカウンタ値が「2」になったとき、すなわちRTCを検出したときにはプリンタ・ライン・ビット・プレーン・ブラック・イグジット・フラグPLNBLEKが「1」か否かを判別して、「1」であれば上述したように黒(1)があるのでイメージデータ(最終のイメージデータである)をプリンタに出力した後、また「1」でなければ全て白であるのでそのままヘッドコントロールコマンド「FF」(改ざり)をプリンタに出力する。また、プリンタ・ライン・ビット・プレーンカウンタPLCNT

のカウンタ値が最大値「MAX」になったときにも当該用紙には最早印字できないので、ヘッドコントロールコマンド「FF」(改ざり)をプリンタに出力する。

次に、この作成処理におけるビットセット処理について第1図を参照して説明する。

このビットセット処理においては、まずライン・ビット・ポイントLINBP及びライン・水平・ポイントLINHPで示すビットのセット指示を受ける。

そして、ライン・水平・ポイントLINHPのビットD₀、D₁、(b₀、b₁)で示されるプリンタ・ライン・ビット・プレーンPLBPを構成する3個のパツファの内のいずれかのパツファ(第14図参照)を選択し、ライン・ビット・ポイントLINBPと選択したパツファから実アドレスを得る。

その後、ライン・水平・ポイントLINHPのビットD₀、D₁、D₂からビット位置を得て、このビット位置に黒(1)をセットす

る。なお、このとき、印字ビットが3.85ライン/mmのプリンタでは、ビットD₀、D₁、D₂+1のビットについても黒を配置して同じドットラインのデータを2ドットライン分生成する。

次に、プリンタへの出力処理について第17図を参照して説明する。このプリンタ出力処理タスクはデコード処理タスクからのメッセージ(例えば「メッセージプリンタコマンドMSGPRC」とする)を受けて起動され、処理の終了時にデコード処理タスクに対してメッセージ(例えば「メッセージプリンタレスポンスMSGPRR」とする)を返送するものとする。

このプリンタ出力処理では、まずプリンタがノットビジー状態(レディ状態)か否かを判別して、プリンタがノットビジー状態でなく使用中であればブザーを鳴してメッセージプリンタレスポンスMSGPRRとして「エラー」を返送する。

これに対して、プリンタがノットビジー状態であれば、プリンタに対して初期化コマンド「ESC, R」を送出して、プリンタの初期設定(6L

PI, ホームポジション, SOCP, 品とする) をさせる。

そして、プリンタに出力するデータがイメージデータか否かを判別し、イメージデータであれば、イメージデータコマンド「ESC, %, 1, n1, n2」(n1, n2でイメージデータの長さを示す)を出力して、イメージデータを出力してイメージデータを印字させ、キャリッジリターンコマンド「CR」及びラインフィードコマンド「LF」を出力して印字ヘッドを次行先頭位置に位置させた後、メッセージプリンタレスポンスMSGRとして「レディ」を返送し、メッセージプリンタコマンドMSGRCを受けたときに再度イメージデータか否かの判別処理に戻る。

また、プリンタに出力するデータがイメージデータでなければ、改行コマンド「LF」か否かを判別し、改行コマンド「LF」であれば、改行コマンド「LF」を出力した後、メッセージプリンタレスポンスMSGRとして「レディ」を返送し、メッセージプリンタコマンドMSGRC

を受けたときに再度イメージデータか否かの判別処理に戻る。

さらに、プリンタに出力するデータが改行コマンド「LF」でなければ、改行コマンド「FF」か否かを判別し、改行コマンド「FF」であればキャンセルコマンド「CAN」及び改行コマンド「FF」を出力した後、メッセージプリンタレスポンスMSGRとして「レディ」を返送してこの処理を終了し、また改行コマンド「FF」でなければメッセージプリンタレスポンスMSGRとして「エラー」を返送してこの処理を終了する。

次に、受信データ(圧縮コード)に基づいて用紙サイズを判定する用紙サイズ判定処理について第18図乃至第20図を参照して説明する。

まず、この実施例で使用している10"プリンタでは、1ラインのドット数が前述したように、1440程度であり、実際には

A4サイズ…1862ドット

B4サイズ…1440ドット

を使用している。つまり、第18図に示すように、B4サイズでは1440ドット全部を使用するが、A4サイズでは両端の各々39ドット(合計78ドット)は使用しない。

また、前述したようにファクシミリ装置では1ラインが1728ドットであるのに対してプリンタは1ラインが1440ドットであり、第18図に示すように送信時に両端に各々144ドットの白ランを付加するので、受信FAX面情報の内の両端の144ドットはプリンタの1ラインをはみでることになる。

そこで、FAX面情報(圧縮コード)を受信してデコードするときに、各ラインの最初の白ランの長さの内での最小値と、各ラインの最後の黒ラインの位置の内での最大値とを検出し、この検出した最小値が「144+39=183」ドット位置より小さいとき、又は最大値が「144+39+1362=1545」ドット位置より大きいときにはB4サイズと判定し、そうでなければA4サイズと判定してFAX面情報と共に蓄積して、

印刷時にその判定したサイズを提示するようにしている。

この処理について第20図を参照して説明すると、まず左側の最小白ランポジションを格納するレジスタLPOS及び右側の最大黒ランポジションを格納するレジスタRPOSを「0」にセット(クリア)した後、受信データ(圧縮コード)からEOLをサーチする。

その後、1ライン中のデコード結果のランポジションをカウントするためのカウンタRUNCNT及び1ライン中のデコード結果の黒の右側の最大ポジションをカウントするためのカウンタLRUNCNTを「0」にセット(クリア)した後、受信データをデコードしてランレングスのビット数を求める。なお、これ等のカウンタRUNCNT及びカウンタLRUNCNTのカウント値は1ライン中でのテンポラリ値である。

そして、ランレングスのビット数(以下「RUN」と称する)がレジスタLPOSの値より小さい(RUN<LPOS)か否かを判別して、RU

$N < LPOS$ であればRUNをレジスタLPOSにセットする。その後、RUNにカウンタRUNCNTの値を加算した値を再度カウンタRUNCNTにセットする。

その後、次の受信データをデコードし、EOLか否かを判別する。このとき、EOLでなければ、RUNにカウンタRUNCNTの値を加算した値を再度カウンタRUNCNTにセットした後、黒ランか否かを判別し、黒ランであればカウンタRUNCNTの値をカウンタLRUNCNTにセットした後、また黒ランでなければそのまま、次の受信データのデコード処理に戻る。

そして、受信データがEOLになったときには、カウンタRUNCNTの値が「1728」か否かを判別し、カウンタRUNCNT=1728であれば、カウンタLRUNCNTの値がレジスタRPOSの値より大きい($LRUNCNT > RPOS$)か否かを判別し、 $LRUNCNT > RPOS$ のときにはカウンタLRUNCNTの値をレジスタRPOSにセットする。これに対して、カウン

タRUNCNT=1728でなければエラー処理をする。

その後、1頁が終了したか否かを判別して、1頁が終了していなければカウンタRUNCNTを「0」にクリアする処理から再度上述した処理を行なう。このようにして、レジスタLPOSには1頁の各ラインの内での白ランポジションの最小値を格納し、レジスタRPOSには1頁の各ラインの内での黒ランポジションの最大値を格納する。

そして、1頁が終了したときには、レジスタLPOSの値が「183」(144+39)より小さい($LPOS < 183$)、つまり左側の最小白ランポジションが「183」より小さいか否かを判別し、レジスタLPOSの値が「183」より小さいときにはB4サイズと判定し、またレジスタLPOSの値が「183」より小さくなければレジスタRPOSの値が「1545」(144+39+1362)より大きい($RPOS > 1545$)、つまり右側の最大黒ランポジションが「1545」より大きいか否かを判別し、レジスタR

POSの値が「1545」より大きいときにはB4サイズと判定し、更にレジスタRPOSの値が「1545」より小さくなければA4サイズと判定する。

なお、この処理は相手先が自己と同等の通信制御装置であつて180DPIで印字データを送ってくる場合の例である。

そして、このようにして得られた用紙サイズを印刷時に表示することによつてオペレータに適切な用紙のセットを促し、あるいは複数サイズの用紙をセット可能な例えばオートシートフィーダを格納したプリンタであれば自動的に適切な用紙を選択することによつて、送信側に応じた適切な用紙を使用してプリンタで印刷することができ、データの欠落等を生じない。

なお、上記実施例においては、ホスト側からプリンタに対する印字データがイメージデータで転送される例について述べたので、通信制御装置自体には文字コードを文字パターン(イメージデータ)に変換するキャラクタジェネレータを備えて

いないが、ホスト側からの印字データが文字コードで送られてくるときには通信制御装置内にキャラクタジェネレータを備えればよく、またこの内部キャラクタジェネレータの使用を選択する選択スイッチを設ければホスト側がイメージデータ及び文字コードのいずれを使用するものであつても接続できる。

また、上記実施例においては、この発明をプリンタに接続してファクシミリ装置から受信する通信制御装置について述べたが、これに限らず圧縮コードを受信するすべての通信端末装置に実施することができる。

さらに、プリンタの構造も上記実施例で述べたものに限られないことも勿論である。

効果

以上説明したように、この発明によれば、適切な用紙を使用することができる。

4. 図面の簡単な説明

第1図はこの発明を実施した通信制御装置の要部を機能的に示すブロック図。

第2図は同じくその通信制御装置を解えた情報処理システムの一例を示す構成図。
 第3図は同じくその通信制御装置の具体的構成を示すブロック図。
 第4図は同じくホスト側から入力されるコマンドを解析するコマンド解析処理を示すフロー図。
 第5図及び第6図は同じくホスト側からのデータ転送の説明に供する説明図。
 第7図及び第8図は同じくワークメモリへのビット・マップ展開の説明に供する説明図。
 第9図は同じくそのワークメモリの説明に供する説明図。
 第10図は同じくビット・マップ展開処理の一例を示すフロー図。
 第11図及び第12図は同じくラインチェック処理の一例を示すフロー図及びその具体的説明に供する説明図。
 第13図及び第14図は同じくデコード処理の説明に供する説明図。

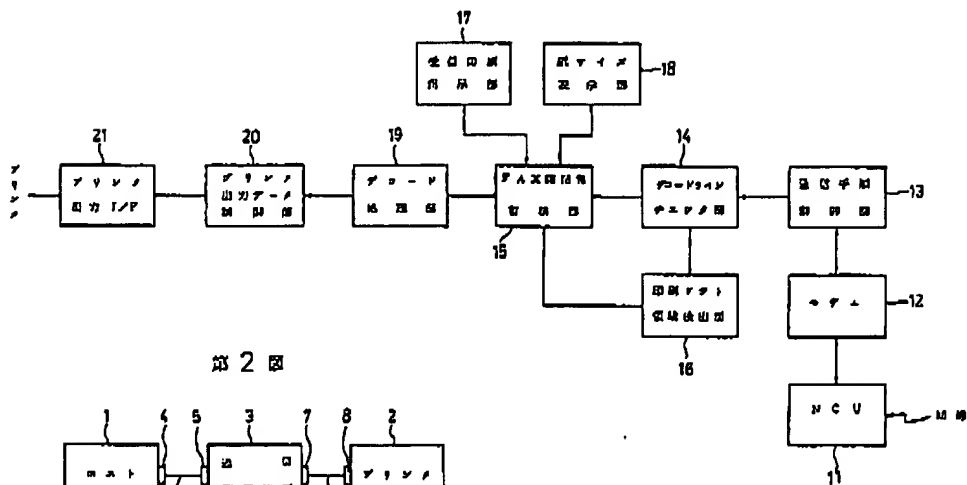
第15図乃至第17図は同じくデコード処理、ビットセット処理及びプリンタ出力処理を示すフロー図。
 第18図及び第19図は同じく用紙サイズ判定処理の説明に供する説明図。
 第20図は同じく用紙サイズ判定処理の一例を示すフロー図である。

3…通信制御装置 13…通信手順制御部
 14…デコードラインチェック部
 15…PAX面情報格納部
 16…印刷ドット領域検出部
 17…受信印刷指示部 18…紙サイズ表示部
 19…デコード処理部
 20…プリンタ出力データ制御部
 21…プリンタ出力インタフェース

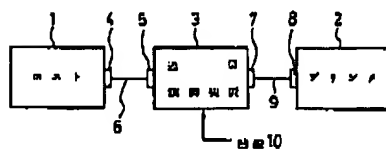
出願人 株式会社 リ コ ー
 代理人 井理士 大 塚 敏



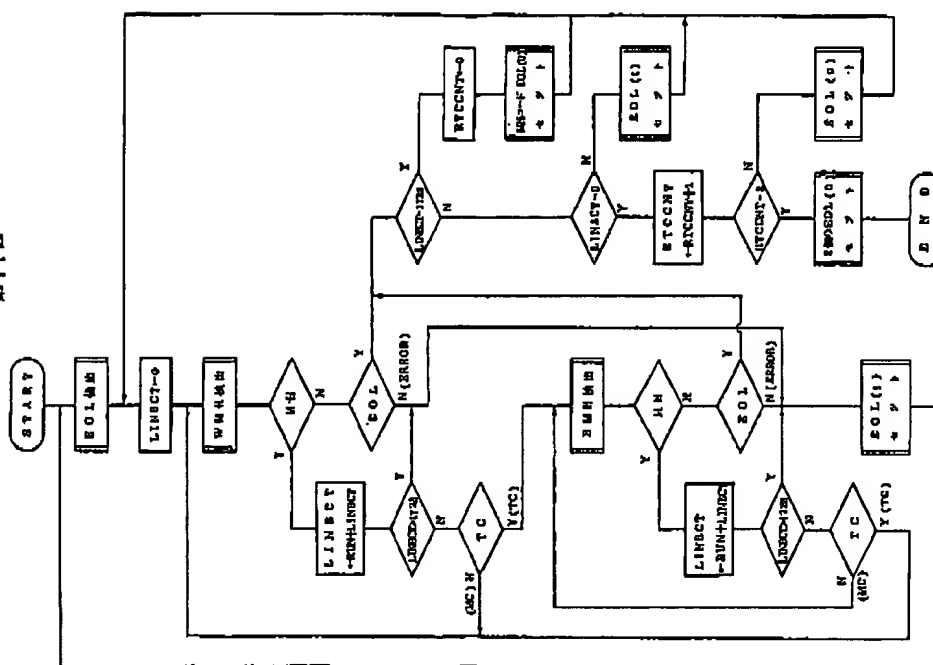
第1図



第2図



第一區



第13圖

[illegible]

第14図

| | | | | | |
|------|-----------------|----------|---------------------|----------|---------------------|
| 0 | 0 | 1718 | 1 | 1723x2 | 2 |
| | 3 | | 4 | | 5 |
| | | | | | |
| 143 | 143×3 | | $143 \times 3 + 1$ | | $143 \times 3 + 1$ |
| 144 | 144×3 | | $144 \times 3 + 1$ | | $144 \times 3 + 2$ |
| | | | | | |
| 1583 | 1583×3 | | $1583 \times 3 + 1$ | | $1583 \times 3 + 2$ |
| 1584 | 1584×3 | | $1584 \times 3 + 1$ | | $1584 \times 3 + 2$ |
| | | | | | |
| 1727 | 1727×3 | 1728x2-1 | $1727 \times 3 + 1$ | 1728x3-1 | $1727 \times 3 + 2$ |

5184

第15図

